CLAIMS

[0018] What is claimed is:

1. A loop detector comprising a set of entries and an array to store a set of loop iterations, wherein the number of entries in said array is smaller than the number of entries in said loop detector.

2. The loop detector as in claim 1, wherein entries in said array are fully associative.

3. The loop detector as in claim 1, wherein the set of entries in said array stores a counter to count speculative iterations of said loop.

4. The loop detector as in claim 1, wherein the set of entries in said array stores a counter to count real iterations of said loop.

5. A method of storing a counter of loop iterations, the method comprising:

    determining if loop iteration data for a branch is stored in an entry of an array, said array associated with a loop detector; and

    incrementing a counter in said array entry;

    wherein the number of entries in said array is smaller than the number of entries in said loop detector.

6. The method as in claim 5, comprising copying a number of actual iterations of said loop into said array entry.

7. The method as in claim 5, comprising allocating an entry of said array based on the least recently used entry in said array.

8. The method as in claim 5, wherein said incrementing said counter in said array comprises incrementing a counter of actual iterations of said loop.

9. The method as in claim 5, wherein said incrementing a sum in said array comprises incrementing a counter of speculative iterations of said loop.

10. The method as in claim 5, wherein the entries of said array are fully associative.

11. A method of counting loop iterations, comprising storing a counter of loop iterations in an array, wherein entries in said array are associated with more than one entry in a loop detector.

12. The method as in claim 11, wherein storing a counter of loop iterations comprises storing a counter of speculative loop iterations.

13. The method as in claim 11, wherein storing a counter of loop iterations comprises storing a counter of real loop iterations.

14. The method as in claim 11, comprising allocating a branch with an entry in said array based on a least recently used entry in said array.

15. The method as in claim 11, wherein storing a counter of loop iterations in an array comprises storing a counter of loop iterations in a fully associative array.

16. A processor comprising a loop detector, said loop detector comprising an array to store a counter of loop iterations, wherein entries in said array are capable of being associated with more than one entry in said loop detector at various times.

17. The processor as in claim 16, wherein said counter of loop iterations is a speculative counter of loop iterations.

18. The loop detector as in claim 16, wherein entries in said array are fully associative.

19. A system comprising:

a dynamic random access memory unit; and

a processor comprising a loop detector, said loop detector comprising an array to store a counter of loop iterations, wherein entries in said array are capable of being associated with more than one entry in said loop detector.

20. The system as in claim 19, wherein said counter is to count speculative iterations of said loop.

21. The loop detector as in claim 19, wherein said counter is to count real iterations of said loop.